A Sketch of
Multiresolutional Decision Support Systems Theory

A. Meystel

Department of Electrical and Computer Engineering, Drexel University
Philadelphia, PA 19104

Abstract

Multiresolutional Decision Support Systems gain better performance and higher accuracy by the virtue of building highly efficient multiresolutional representation and employing multiscale Behavior Generation Subsystem (Planning and Control). The latter are equipped by devices for unsupervised learning that adjust their functioning to the results of self-identification. We show planning and learning to be joint processes.

Keywords: *behavior generation, control, decision support, generalization, knowledge, learning, instantiation, multiresolutional, multiscale, planning, randomized, representation, resolution, search*

1. Introduction

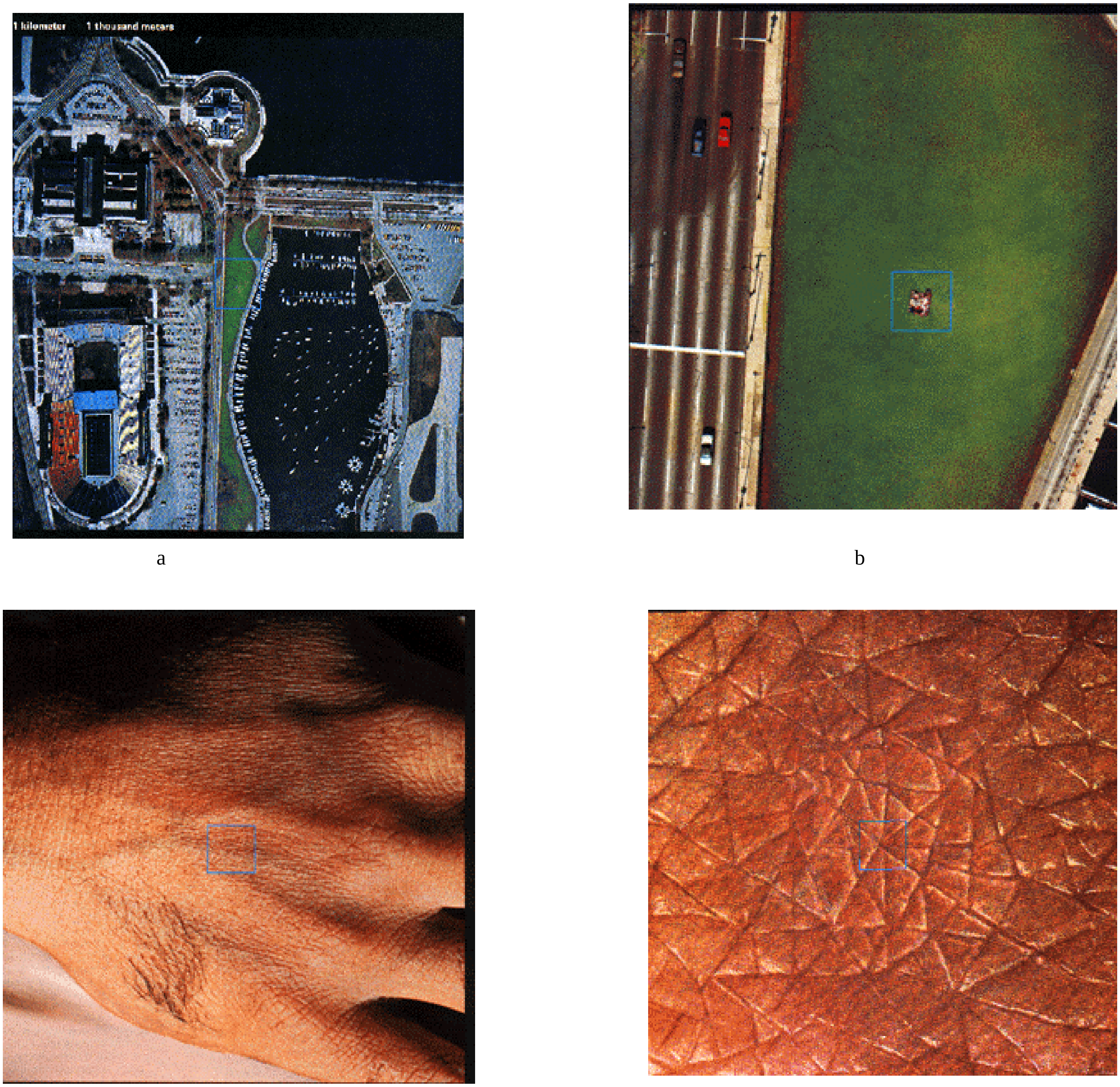
Multiresolutional Representation (MR) of the World should be considered one of the tools in the arsenal of Knowledge Management [1]. It is the tool that is widely applied but is scarcely noticed, probably because of its overwhelming omnipresence. The concept of MR can be illustrated by the series of pictures shown in Figure 1 (a - f). The enhanced set of these pictures with much more details can be seen in [2]. The resolution of each subsequent image is increased by an order of magnitude while the are of observation is simultaneously reduced by two orders of magnitude. It is not difficult to deduce that as far as the underlying knowledge is concerned, the objects in the image *f* are contained in the image *e* [(skin texture)⊃(hand)], the objects of the image *e* are contained in the image *d* [(hand)⊃(sleeping person)], the objects of the image *d* are contained in the image *c* [(sleeping person)⊃(picnic)], the objects of the image *c* are contained in the image *b* [(picnic)⊃(green lawn)], and the objects of the image *b* are contained in the image *a* [(green lawn)⊃(part of the city)].

This MR nestedness of sub-processes and sub-systems of the overall processes and system in not obvious in a standard cursory analysis, it can be discovered only as a result of special observations (computer vision equipment) and investigative analysis. More importantly, it is not obvious that this nestedness of entities and their properties is important (if necessary at all) for supporting the decision making activities at each level of resolution. Yet, all images in Figure 1 are tightly linked by the prior cognitive activities that are unified by identical processes of generalization performed upon higher resolution images to obtain a lower resolution image. Similar processes of instantiation allow for receiving each higher resolution image from the lower Resolution. Actually, not the process of sensing or the process of image edges detection and segmentation (and others) determine further image understanding and interpretation but rather the joint processes of generalization and instantiation that are executed upon these images top-down and bottom-up.

As the signals measuring and processing is conducted, at each particular level of resolution they contain a different package of frequencies (Figure 2). It demonstrates that the granularity of representation is correlated with the bandwidth of the signals at a level.

Why do we encounter this phenomenon: multiresolutional knowledge representation? Why the mechanisms emerged of generalization and instantiation? The reduction of complexity via reduction of "multiplicity" could only be done by the virtue of grouping and representing the group by a single symbol. This semiotic principle emerged because of the need to reduce computational burden. Computational benefits for a particular example of knowledge representation associated with planning is given in [8, 9].

The system of representation based upon recursive grouping/decomposition incorporates and uses the algorithms of generalization and instantiation in different incarnations that depend on circumstantial factors as for example, the information we are dealing with, or the subsystem of the world where the results are applied. Thus, the learning system must employ the same tools: labeling the entities in order to deal with concise notations (symbols), grouping the entities, decomposing them if information details are required. Learning systems use the same computational mechanisms.



a b

e f

Figure 1. Consecutive increasing the resolution of representation

The decision support system (DSS) treats knowledge as an entity suggested in [1]: it employs the awareness of familiarity gained by experience for storing experiences as well as for constructing decisions (including plans and controls) that ensure functioning of a goal-oriented system with increased performance index. MR gives an opportunity to minimize the value of computational complexity in a subset of DSS that organizes knowledge by joint processes of generalization and instantiation and use nested MR-search for converging to a recommended solution. This concept was introduced for planning and control purposes in 1986 [2] and explored in depth in subsequent works [3-10].

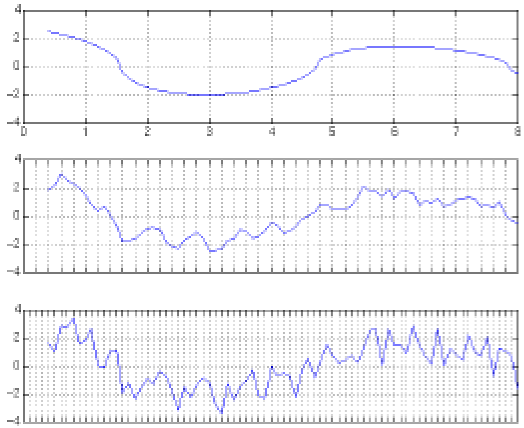


Figure 2. Multiresolutional representation (a-low resolution, b-mid-resolution+low resolution, c-the sum of signals of all levels)

All these processes dwell on the processes of learning employed by MR-DSS.

2. Decision Support of Behavior Generation

The structure of power station control system shown in Figure 3 was successfully tested at Delmarwa Power Station, DE, USA [11]. is required in all faculties of a system shown in Figure 3. Three levels of resolution are demonstrated Low ("Task Level"), Middle ("Component Level") and High ("Actuation Level"). Each level forms a loop closed through connection **1**. Each of these loops is a loop of "closure" [9] and is equivalent to the Elementary Loop of Functioning (ELF) described in [6, 7, 8].

The vertical subsystem 5 (Plant) from Figure 3 is equivalent to subsystem Sensors, World and Actuators from the Elementary Loop of Functioning that is described in [8]..

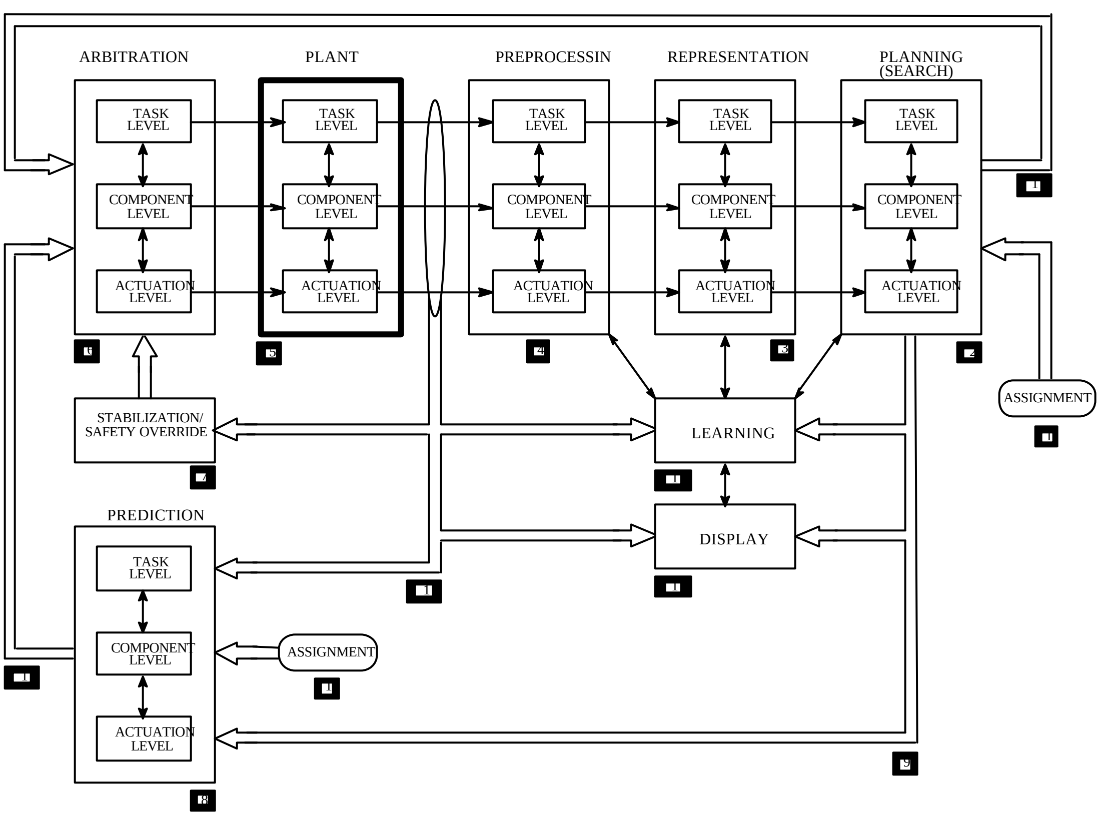


Figure 3. Architecture of the Power Station Multiresolutional Decision Support System for energy efficient Planning/Control

Figure 3 contains several subsystems that should be added to the standard ELF if there is a need to equip the system by Multiresolutional Decision Support: these are the systems of Learning and Prediction. Learning enhances Representation, while Prediction, together with Stabilization arrives at the subsystem of Arbitration (6). Standard ELF is the core structure demonstrating the important property of any functioning system, including Intelligent Systems: it shows the property of *closure*. The meaning of closure is in the fact that the proper functioning requires the loop of information flows to be closed.

What happens in the subsystem of Behavior Generation? The latter has two mechanisms: 1) maintenance of the ontology that organizes the inventory of the known symbols and their definitions and keeps relationships of nestedness with ontology subsystems of other resolution levels, 2) combinatorial search engine that performs *planning* i. e. creates alternatives of imaginary (possible and desirable) worlds, and 3) *simulator* engine that explores expected behaviors of the alternatives of the imaginary worlds and/or monitors the *execution* processes.

The goal of functioning to be achieved by the system arrives at the subsystem of Behavior Generation [9] that is equipped by mechanisms of planning and execution. At the present time, these mechanisms cannot be considered as thoroughly studied, and the general theory of planning can hardly be attempted. We will discuss a subset of problems in which the goal is defined as the attainment of a particular state or a particular string of states. Other types of problems can also be imagined: in chess the goal is clear -to win but this goal demand achieving a special configuration (mate-situation) but it cannot be achieved by arriving at a particular pre-determined position in a space (even in a descriptive space.) Most of the problems related to the theory of games and linked with pursuit and evasion are characterized by a similar predicament and are not discussed here.

Planning is understood as searching for appropriate future trajectories of motion leading to the goal. Searching is performed within the system of representation (simulation) that gives a tremendous advantage in comparison with searching via trying.

3. Planning in a Representation Space with a Given Goal

The world is assumed to be judged upon by using its Space of Representation, or its State Space which is interpreted as a time tagged vector space with a number of properties. Any activity in the World (State Space or Representation Space) is called *motion*. It can be characterized by a trajectory of motion with the “working point” or “present state” (PS) that is traversing the space from one point (initial, or state, IS) to one or many other states (goal states, GS.) The goal states are given initially from the external source as a “goal region”, or a “goal vicinity” in which the goal state may not be completely defined in a general case. This vision of the problem of Behavior Generation was dominating in the area of Control Systems. Planning was unified with Control only recently when it became clear that both Planning and Control are involved into anticipation of the preferable motion (off-line) with some appropriate correcting activities (on-line).

One of the stages of planning (often the initial one) serves for defining where exactly is the GS within the “goal region.” In this paper, we will focus upon a subset of planning problems where one or many GS remain unchanged through all period of their achievement. Traversing from IS to GS is associated with consuming time, or another commodity (performance index, or cost.).

Planning Problems in Behavior Generation is frequently associated with the domain of robotics or automated control systems although it is absolutely equivalent to planning in all other domains. Robotics became the integrated research domain that provided for blending the goals and testing the means of achieving them, i.e. a domain with a direct need for planning. In 1983, T. Lozano-Perez has introduced the idea of search in "configurations space". From the experience of using this search, it became clear that the exhaustive search would be computationally prohibitive if the configuration space is tessellated with the final accuracy required for motion control. The theory of configuration space made one important thing obvious: planning is searching for admissible alternatives. This development helped to realize that planning should combine the exhaustive (often meaningfully complex) search off-line, and an efficient algorithm of an off-line control.

At this period of time the engineering community stopped talking about control of actions and introduced a more balanced term of Behavior Generation

The overview of the situation in the area of planning and control can be found in [9]. The recommended algorithm should be aligned with the following suggestions. Consider the Ω state space in which the start and final points SP and FP are given. The minimum cost path from SP to FP is to be found with the final accuracy ρ. Let us consider particular cases $\Omega=\Omega_1$ and $\rho=\rho_m$. To declare the final accuracy is equivalent to applying some mechanism od space tessellation. One of mechanisms of tessellation is distributing discrete points in the state space. We will distribute them in a random fashion and then, will determine the minimum cost path while considering these points as vertices of an imaginary graph. The condition of constructing random tessellation reflects uncertainties of the system that should be available for evaluation from the existing representation. In Figure 4,a the random points are distributed in the state space with obstacles. An example of the result of running a minimum-cost algorithm in the tessellated state space with obstacles is shown in Figure. 4,b.

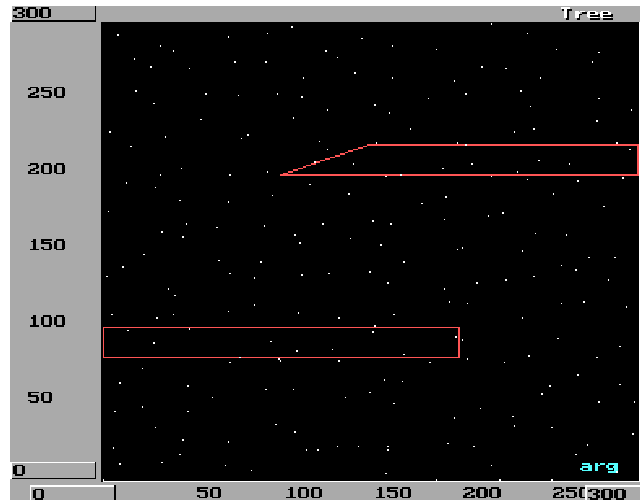
Since the graph is randomized, the trajectory is a random one, too. If one runs the search algorithm a number of times, we receive the results of searching as a "stripe of solution" as shown in Figure 5,a. Then, we get a privilege to continue with constructing tessellations of higher resolution only within this stripe as shown in Figure 5,b. Then, when we run the minimum cost search-algorithm only within this "finely" tessellated stripe with high resolution density of tessellata, we receive a high resolution plan. This process can be repeated recursively within as many levels as necessary.

We will introduce three operators that describe the above computations.

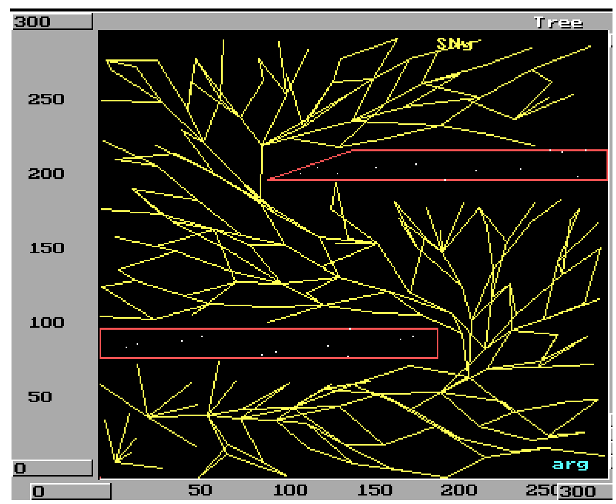
I. Operator of Representation (Â)

$$\Re:(\Omega,\rho)\rightarrow M, \text{ or } M=\Re(\Omega,\rho) \quad , \qquad (1)$$

where M- is the map representing the state-space Ω, ρ is the level of resolution of this map determined by the density of the search-graph that we intend to run at this particular level of resolution (determined by the accuracy ρ). This is a non-trivial operator because it presumes discovery of entities, putting them into relationships with each other, generalization, instantiation and measuring relationships (including costs).



a



b

Figure 4 Randomized tessellation of the state space (a) and a single running of the minimum cost trajectory algorithm (a rectangle and a trapeze – obstacles)

II. Operator of state space search (S^3)

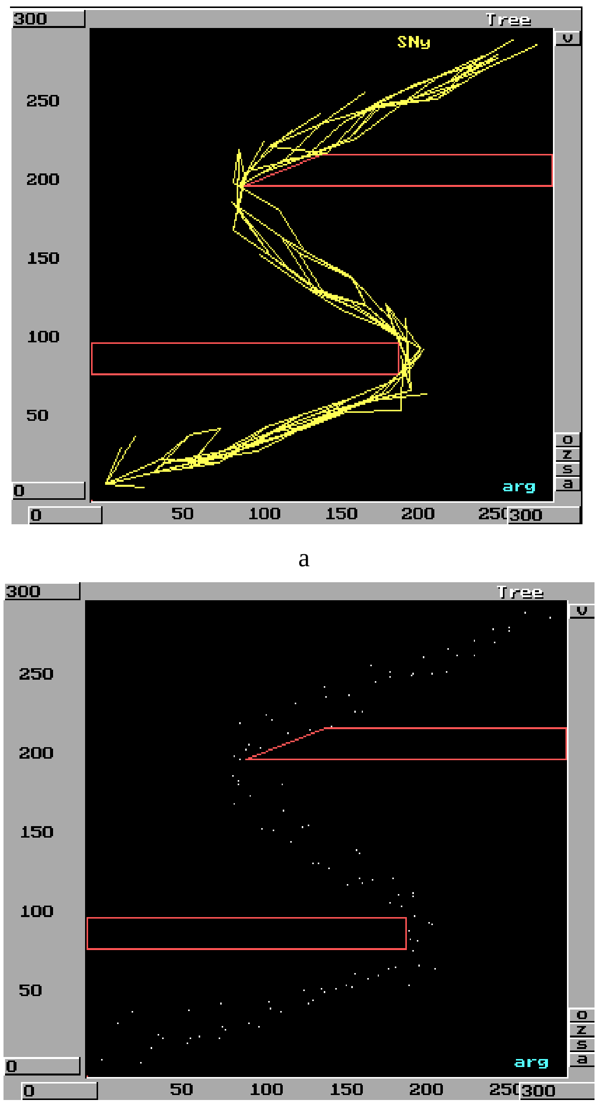
$$S^3: (M, SP, FP, J, \rho)\rightarrow P \quad , \qquad (2)$$

or $P=S^3(M)$,where P- is the optimum path connecting the start point SP and the finish point FP with tessellation constructed for the accuracy ρ. J- is the cost of operation which should be minimized as a result of search S^3. This operator should be based upon one of the minimum-cost algorithms (e.g. Dijkstra) and tailored for specifics of the problem.

III. Operator of space contraction (C) which determines the width of stripe and the new final goal for the algorithm of search.

$$C:(P, w)\rightarrow\Omega, \text{ or } \Omega=C(P), \qquad (3)$$

where w- is the width of the "stripe" obtained after several runs of the search algorithm. Instead of "stripe" one can use the term "envelope", (e.g. the "width" is the parameter of the envelope).



b

Figure 5 Multiple running of the minimum cost trajectory algorithm (a) and the uncertainty stripe obtained as a result of the multiple running (b)

The hierarchical control algorithm can be described as follows:

for k=1, ..., m do the following string of procedures:

a) $\Omega_k = C(P_{k-1})$, or at k=1 assume $\Omega_k = \Omega$,

b) $M_k = \Re(\Omega_K, \rho_k)$,

c) $P_k = S^3(M_k)$.

The algorithm of control can be represented as a diagram

$$\begin{tikzcd} & w \arrow[d] & \rho_k \arrow[d] & SP, FP, J \arrow[d] & \\ P_{k-1} \arrow[r] & X \arrow[r] & P \arrow[r] & \Sigma^3 \arrow[r] & P_k \\ & & \Omega_k \arrow[u] & M_k \arrow[u] & \end{tikzcd} \quad (4)$$

or a recursive expression

$$P_k = S^3(R\ (C\ (P_{k-1}, w), \rho_k\ )\ SP, FP, J) \qquad (5)$$

The algorithm (5) has proven to be good for off-line search in the state space. In the class of on-line problems the process of control is to be described by the trajectory of "working point" moving in the state space.

4. Learnable Representations

All Representation Spaces are acquired from the external reality by the processes of Learning. Many types of learning are mentioned in the literature (supervised, unsupervised, reinforcement, dynamic, PAC, etc.). We will focus primarily on processes of unsupervised learning [12]. Before classifying the needs for a particular method of learning and deciding how to learn, we would like to figure out what should we learn. Now, it is not clear whether the process of learning can be separated algorithmically into two different learning processes: a) of *objects representation*, and b) of the *rules of action representation*, or are these two kinds of learning just two sides of the same core learning process. In both cases, learning is storing and generalizing information of experiences with their values associated with achieving particular goals.

The following knowledge should be contained in the Representation Space. If no GS is given, any pair of *state* representations should contain implicitly the *good rule* of moving from one state to another. In this case, we consider any second state as a provisional GS.

We will call "proper" representation a representation similar to the mathematical function and/or field description: at any point of the space, the derivative is available together with the value of the function. The derivative can be considered an action required to produce the change in the value of the function.

We will call "goal oriented" representation a representation in which at each point a value of the action is given required for describing not the best way of achieving an adjacent point but the best way of achieving the final goal. Both "proper" and "goal oriented" representations can be transformed in each other. Neither is mandatory for functioning: valued *memories of experiences* (ME) are sufficient.

5. The Artifacts of Representation Space: The Phenomenon of "Sea Weeds"

Representation as sets of valued ME is characterized by the following artifacts:

- existence of states with its boundaries determined by the resolution of the space each state is presented as a tessellatum [9], or an elementary unit of representation, the smallest discernible unit of attention)
- characteristics of the tessellatum which is defined as an indistinguishability zone; we consider that resolution of the space shows how far the "adjacent" tessellata (states) are located from the "present state" (PS)
- lists of coordinate values at a particular tessellatum in space and time
- lists of actions to be applied at a particular tessellatum in space and time order to achieve a selected adjacent tessellatum in space and time
- existence of strings of states intermingled with the strings of actions to receive next consecutive tessellata of these strings of states
- boundaries (the largest possible bounds of the space with similar properties, i. e. the obstacles
- costs of traversing from a state to a state and through strings of states.

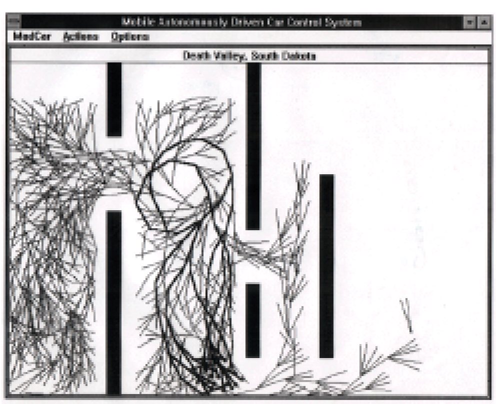


Figure 6. The sea weeds

When ME are clustered into classes of similarity (e.g. of adjacency) they remind visually masses of "sea weeds". In many cases, the states contain information pertaining to the part of the world beyond our ability to control it, and this part is called "environment." The part of the world to be controlled is the system for which we plan often referred to as "self." Thus, the representation is a part of "self" including knowledge about actions that "self" should undertake in order to traverse the environment. Plans are formed as strings of preferable "sea weeds" combined together.

6. Planning in Redundant Systems

Non-redundant systems have a unique trajectory of motion from one state to another. Redundant systems are defined as systems with more than one "the best" trajectory of motion from initial (IS) to final states (FS)

These systems contain a **multiplicity of alternatives of space traversal**. Redundancy grows when the system is considered to be a stochastic one. The number of available alternatives grows even higher when we consider also a multiplicity of goal tessellata at a particular level of resolution. This happens when the goal is being assigning at a lower resolution level which is the fact in multiresolutional systems (such as NIST-RCS [8, 9])

In non-redundant systems, there is no problem of planning. The problem is to find the unique trajectory and to provide tracking of it by an appropriate classical control system.

7. Learning as a Source of Representation: Storing and Clustering "Sea Weeds"

Learning is defined as knowledge acquisition via experience of functioning. Thus, learning is development and enhancement of the representation space. The latter can be characterized in the following ways:

- by a set of paths (to one or more goals) previously traversed
- by a set of paths (to one or more goals) previously found and traversed
- by a set of paths (to one or more goals) previously found and not traversed
- by a totality of (all possible) paths
- by a set of paths executed in the space in a random way.

One can see that this knowledge contains implicitly both the description of the environment and the description of the actions required to traverse a trajectory in this environment.

All information arrives in the form of experiences. The "learned" representation as a set of strings of valued "sea weeds" is equivalent to the multiplicity of explanations how to traverse, or how to move.

8. Types of Problems of Planning

Any problem of planning is associated with

- actual existence of the present state
- actual, or potential existence of the goal state
- knowledge of the values for all or part of the strings of executable states as far as some particular goal is concerned.

Any problem of planning contains two components: to refine the goal (bring it to the higher resolution) and to determine the path to this refined goal. They are performed together, or separately and can be formulated as follows:

a) given PS, GS and KS find the subset of KS with a minimum, or a prearranged cost, or with a cost in a particular interval.

b) given PS, Gs from the lower resolution level and KS (all paths) find the GS with a particular value

Finding solutions for these problems is done by a process of *planning*. In other words, planning is a construction of the goal states, and/or strings of preferable states connecting the present state with the goal states. There is a striking similarity and interrelatedness between *planning* and *learning*, actually their inseparability.

In order to do this, we must learn where the goal is located by consecutive refinement of the initial coarse information. In all cases it is associated with reduction of the indistinguishability zone and the size of the tessellatum associated with a particular variable, i.e. the accuracy of representation grows. We plan and learn by testing: in the representation, for planning, and in the reality, for learning. Learning via testing simulated systems is becoming more and more wide spread.

The second component is the simulation of all available alternatives of the motion from the initial state. Procedurally, this simulation is performed as a search, i.e. via combinatorial construction of all possible strings (groups). To make this combinatorial search for a desirable group more efficient we reduce the space of searching by focusing attention.

The need in planning is determined by the multialternative character of the reality The process of planning can be made more efficient by using appropriate heuristics which are available via processes of learning.

9. The Unified System of Planning and Learning: A Subsystem of MR DSS

The process of searching for plans is associated either with collection of additional information about experiences, or with extracting from KS the implicit information about the state and moving from state to state, for the purpose of learning. In other words, ***planning is inseparable from and complementary to learning***.

This unified planning/learning process is always oriented toward improvement of functioning in engineering systems (improvement of accuracy in an adaptive controller, improvement of efficiency in energy consuming devices) and/or toward increasing of probability of survival (emergence of the advanced viruses for the known diseases that can resist various medications, e.g. antibiotics.)

This joint process can be related to a system as well as to populations of systems and determines their evolution.

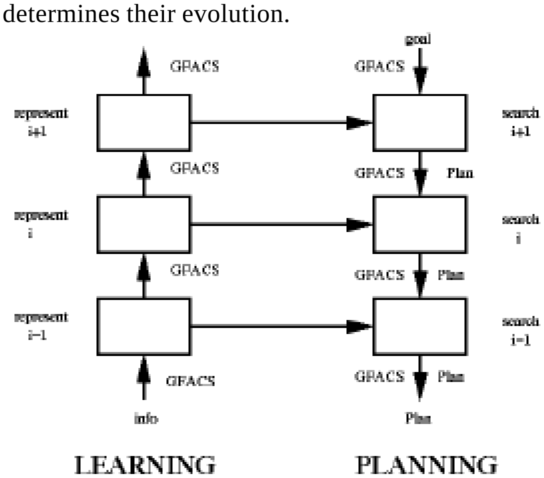


Figure 7. On the relations between planning and learning

10. Planning, Learning, and Control: A Unified Theory

Learning/Planning Automaton. The joint Planning/Learning process is studied by using a tool of Learning/planning automata (LPA) is a tool that allows for jointly exploring these two fundamental processes of intelligent systems. Naturally, it becomes a component of the Multiresolutional DSS.

Elementary Computations. *Search* (S) is always performed by constructing feasible combinations of the states within a subspace ("feasible" means: satisfying a particular set of conditions or constraints.) As many as possible alternatives of feasible motions should be explored and compared. If search is combined with formation of alternatives, we call this procedure *combinatorial search* (CS).

Usually, *grouping* (G) presumes exploratory construction of possible combinations of the elements of space and as one or many of these combinations satisfy conditions of “being an entity”, this group generates a new symbol with subsequent treating it as an object.

The larger the space of search is, the higher is the computational complexity of search. This is why a special effort is allocated with reducing the space of search, i.e. focusing attention (FA) upon reduced sub-spaces. FA results in determining two conditions of searching, namely, its upper and lower boundaries:

a) the upper boundaries of the space where the search is to be performed, the *scope*

b) the resolution of representation (the lower boundaries, the tesselatum)

Via exploring these experiences in planning and learning we arrive at a conclusion that they are always employ these three procedures: grouping, focusing attention and combinatorial search (or subsets of them).

The property of Intelligence. Forming multiple combinations of entities (combinatorial search, CS) satisfying required conditions of transforming them into new entities (grouping, G) within a bounded subspace (focusing attention, FA) is frequently performed as a fundamental set of procedures. Since these three procedures work together we will talk about them as about a triplet of computational procedures (the abbreviations GFACS or CFS are used.) Notice, that in learning it creates lower resolution levels out of higher resolution levels (bottom-up) while in planning it progresses from lower resolution levels to higher resolution levels (top-down). This algorithmic triplet emerges as a tool of multiresolutional representation and/or for the purposes of generating goal-oriented behaviors.

This triplet of computational procedures is characteristic for *intelligence* of living creatures and constructed systems, and probably is the elementary computational unit for

The need in GFACS is stimulated by the property of knowledge representations to intelligence. Its purpose is transformation of large volumes of information into a manageable form that ensures the success of functioning. This explains the pervasive character of hierarchical architectures in all domains of activities including Decision Support Systems.

contain a multiplicity of alternatives of space traversal (i. e. a property of any representations to be **redundant**.) Representations reduce the redundancy of reality. This allows for having problems that can be solved in a closed form (it is a form when no combinatorics is possible and/or necessary).

At each level of resolution, planning is done as a reaction for the slow changes in situation which invokes the need in anticipation and active interference

a) to take advantage of the growing opportunities, or

b) to take necessary measures before the negative consequences occur.

The deviations from a plan are compensated for by the compensatory mechanism also in a reactive manner. Thus, both *feedforward* control (interpreted as planning at all levels of resolution but the highest one) and *feedback* compensation of deviations are reactive activities. Both can be made active in different implementation approaches in control theory.

Examples: a) Classical control systems are systems with no redundancy, they can be solved in a closed form without searching.

b) Any stochastic condition, any type of uncertainty introduced to a control system creates redundancy and requires either for elimination of redundancy or performing search.

c) Optimum control allows for the degree of redundancy that makes searching feasible.

In Figure 8, the process of multiresolutional planning via consecutive search with focusing attention and grouping is demonstrated for the control problem of finding a minimum-time motion trajectory. From Figure 8, one can judge the processes that are performed during the single level S^3-search in the randomized tessellated state space. The reader can identify the processes because a trivial example is considered: minimum time functioning of the dynamic system. The operation won’t change if one is dealing with higher order and/or non-linear system.

The space is learned and encoded in advance by multiple testing, and its representation is based upon knowing that the distance, velocity and time are linked by nonredundant expression. Several methods of constructing attention envelopes are applied.

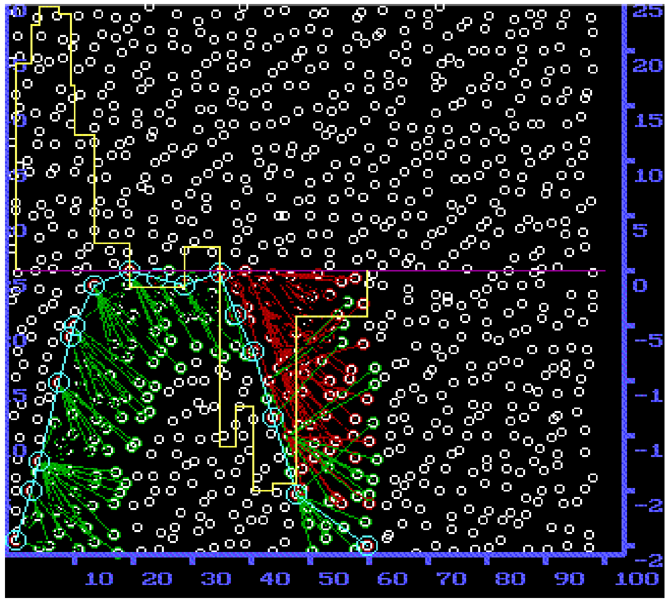


Figure 8. S^3-Search in the state space for the minimum tyme dynamic trajectory

Conclusions

We have demonstrated the advantages of Multiresolutional Decision Support Systems that can be listed as follows:

1. Multiresolutional System of Knowledge Organization allows to reduce complexity and increase efficiency of representation.

2.Most of the Planning/Control problems are being solved via S^3-search (Search in the State Space). The latter requires performing randomized state space tessellation with density of points that reflects the uncertainty of information. Multiresolutional S^3-search allows for stochastic optimization of systems.

3. Representations at each level of resolution are organized as memories of experiences and do not require constructing any analytical model: this system plans and controls with no model required.

4. This representation supports processes of unsupervised learning that contains self-oriented information; no special self-identification is required.

5. The MRDS system was tested in applications to power station energy efficient planning/control system, for planning/control of an unmanned autonomous mobile or spray-casting robots.

References

1. A. Satyadas, U. Harigopal, N. P. Cassaigne, “Knowledge Management Tutorial: An Editorial Overview,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, Vol. 31, No. 4, November 2001, pp. 429-437
2. P. Morrison and P. Morrison, and the Office of C. and R. Eames, *Powers of Ten*, Scientific American Library, A division of HPHLP, NY, 1982
3. A. Meystel, "Planning in a hierarchical nested controller for autonomous robots," Proc. IEEE 25th Conf. on Decision and Control, Athens, Greece, 1986, pp. 1237-1249
4. A. Meystel, "Theoretical Foundations of Planning and Navigation for Autonomous Mobile Systems", *International Journal of Intelligent Systems*, Vol.2, No.2, 1987, pp. 73-128
5. C. Isik, A. Meystel, “Pilot level of a hierarchical controller for an unmanned mobile robot," *IEEE J. of Robotics & Automation*, Vol. 4, No. 3, 1988, pp. 244-255
6. A. Meystel, Autonomous Mobile Robots, World Scientific, World Scientific, Singapore, 1991
7. A. Meystel, “Nested Hierarchical Control,” in Eds. P. Antsaklis, K. Passino, *An Introduction to Intelligent and Autonomous Control*, Kluwer Academic, Boston, 1993, pp. 129-161
8. A. Meystel, “Architectures, Representations, and Algorithms for Intelligent Control of Robots”, in *Intelligent Control Systems: Theory and Applications*, eds. M. M. Gupta, N. K. Sinha, IEEE Press, New York, 1996, pp. 732-788
9. A. Meystel, J. Albus, *Intelligent Systems: Architecture, Design, Control*, Wiley, NY., 2001
10. J. Albus, A. Meystel, *Engineering of Mind: An Introduction to the Science of Intelligent Systems*, Wiley, NY, 2001
11. C. Corson, A. Meystel, F. Otsu, S. Uzzaman “Semiotic Multiresolutional Analysis of a Power Plant”, ”, in Architectures for Semiotic Modeling and Situation Analysis in Large Complex Systems, Proc. of the 1995 ISIC Workshop, Monterey, CA, pp. 401-405
12. A. Meystel, “Learning Algorithms Generating Multigranular Hierarchies”, in *Mathematical Hierarchies and Biology*, Eds.: B. Mirkin, F. R. McMorris, F. S. Roberts, A. Rzhetsky, DIMACS Series in Discrete Mathematics, Vol. 37, American Mathematical Society, 1997